

日 本 国 特 許 庁
JAPAN PATENT OFFICE



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2000年11月24日

出 願 番 号

Application Number:

特願2000-357208

出 願 人

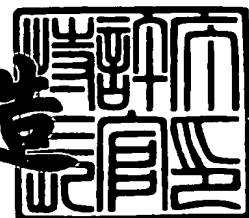
Applicant(s):

ソニー株式会社

2001年 9月26日

特 許 庁 長 官
Commissioner,
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3088299

【書類名】 特許願

【整理番号】 0000661102

【提出日】 平成12年11月24日

【あて先】 特許庁長官 及川 耕造 殿

【国際特許分類】 G06F 1/00
G06F 5/00
G06F 12/00
G06F 13/00

【発明者】

【住所又は居所】 東京都品川区北品川3丁目14番13号 株式会社ソニー
コンピュータサイエンス研究所内

【氏名】 河野 通宗

【発明者】

【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社
内

【氏名】 村田 賢一

【特許出願人】

【識別番号】 000002185

【氏名又は名称】 ソニー株式会社

【代表者】 出井 伸之

【代理人】

【識別番号】 100082762

【弁理士】

【氏名又は名称】 杉浦 正知

【電話番号】 03-3980-0339

【手数料の表示】

【予納台帳番号】 043812

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9708843

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ処理装置および方法

【特許請求の範囲】

【請求項 1】 中央演算処理装置およびメモリからなるデータ処理装置において、

記録媒体の書き込みおよび読み出しを制御するドライバを有し、

コンポーネントから状態保存処理要求があると、上記コンポーネントの依存関係および／または保存データをスナップショットファイルとして上記記録媒体に保存し、

状態復元処理要求があると、上記記録媒体に保存されている上記スナップショットファイルに基づいてコンポーネントの状態を復元するようにしたことを特徴とするデータ処理装置。

【請求項 2】 上記スナップショットファイルは、上記コンポーネント名でタグ付けされていることを特徴とする請求項 1 に記載のデータ処理装置。

【請求項 3】 上記状態保存処理および上記状態復元処理は、上記コンポーネント毎の特定のアドレスに存在する関数を呼び出すようにしたことを特徴とする請求項 1 に記載のデータ処理装置。

【請求項 4】 上記記録媒体は、携帯可能な記録媒体としたことを特徴とする請求項 1 に記載のデータ処理装置。

【請求項 5】 中央演算処理装置およびメモリからなるデータ処理方法において、

ドライバによって記録媒体の書き込みおよび読み出しを制御し、

コンポーネントから状態保存処理要求があると、上記コンポーネントの依存関係および／または保存データをスナップショットファイルとして上記記録媒体に保存し、

状態復元処理要求があると、上記記録媒体に保存されている上記スナップショットファイルに基づいてコンポーネントの状態を復元するようにしたことを特徴とするデータ処理方法。

【請求項 6】 上記スナップショットファイルは、上記コンポーネント名で

タグ付けされていることを特徴とする請求項 5 に記載のデータ処理方法。

【請求項 7】 上記状態保存処理および上記状態復元処理は、上記コンポーネント毎の特定のアドレスに存在する関数を呼び出すようにしたことを特徴とする請求項 5 に記載のデータ処理方法。

【請求項 8】 上記記録媒体は、携帯可能な記録媒体としたことを特徴とする請求項 5 に記載のデータ処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、システムの突発的な電源切断に対応することができるデータ処理装置および方法に関する。

【0002】

【従来の技術】

従来、コンピュータで制御されるシステムに突発的な電源切断が発生した場合、全くの初期状態から起動を行う再起動と、データとして保存しておいた動作中のメモリ状態を全て読み込んで実行を継続するハイバネーションとの何れか一方が行われていた。

【0003】

【発明が解決しようとする課題】

しかしながら、再起動の場合、一般的にデバイスの検出や初期化など全ての処理を行うために時間がかかる問題があった。一方、ハイバネーションの場合、実行しているプログラムや使用中のメモリ量と無関係にメモリ情報を書き出すため、メモリ状態をデータとして保存する時および復帰する時に時間がかかる問題があった。さらに、ハイバネーションは、何らかの処理中にバックグラウンドで実行するようなことはできず、突発的な電源切断に対処することはできない問題もあった。

【0004】

これに対して、瞬電などの突発的な電源切断に対処する方法としては、低電圧状態を検出してアプリケーションに終了処理をさせる方法がある。しかしながら

、これは電圧状態の検出回路が必要であり、また電圧状態の検出回路を設けても、低電圧状態の発生から電源切断までの時間で、全てのデータの保存が可能か否かをシステムソフトウェア側が保証することはできないという問題があった。

【0 0 0 5】

しかしながら、この方法は、あくまで電源切断に対する対策であるため、復帰時については、再起動と全く同様であり、利用可能状態になるまでには、時間がかかる問題があった。

【0 0 0 6】

従って、この発明の目的は、システムに突発的な電源切断が発生しても瞬時に電源切断直前の状態に戻ることができるデータ処理装置および方法を提供することにある。

【0 0 0 7】

【課題を解決するための手段】

請求項 1 に記載の発明は、中央演算処理装置およびメモリからなるデータ処理装置において、記録媒体の書き込みおよび読み出しを制御するドライバを有し、コンポーネントから状態保存処理要求があると、コンポーネントの依存関係および／または保存データをスナップショットファイルとして記録媒体に保存し、状態復元処理要求があると、記録媒体に保存されているスナップショットファイルに基づいてコンポーネントの状態を復元するようにしたことを特徴とするデータ処理装置である。

【0 0 0 8】

請求項 5 に記載の発明は、中央演算処理装置およびメモリからなるデータ処理方法において、ドライバによって記録媒体の書き込みおよび読み出しを制御し、コンポーネントから状態保存処理要求があると、コンポーネントの依存関係および／または保存データをスナップショットファイルとして記録媒体に保存し、状態復元処理要求があると、記録媒体に保存されているスナップショットファイルに基づいてコンポーネントの状態を復元するようにしたことを特徴とするデータ処理方法である。

【0 0 0 9】

コンポーネントから状態保存処理要求が出力されると、コンポーネントの依存関係および／または保存データがスナップショットファイルとして記録媒体に記録される。電源切断が発生したときに、コンポーネントの依存関係に従って、記録媒体に記録されているスナップショットファイルからコンポーネントの状態復元を行う。

【 0 0 1 0 】

【発明の実施の形態】

以下、この発明の一実施形態について図面を参照して説明する。なお、各図に亘り同じ機能を有するものには、同一の参照符号を付し、説明の重複を避ける。図 1 は、この発明が適用されるシステムの一実施形態の概略を示す。この図 1 に示すシステムは、CPU (Central Processing Unit) 1、RAM (Random Access Memory) 2、デバイス 3、不揮発メモリ 4 およびバス 5 から構成される。

【 0 0 1 1 】

RAM 2 は、書き込み／読み出しが自由なメモリである。デバイス 3 は、例えばシリアルデバイスやグラフィックデバイスなどであり、不揮発メモリ 4 は、フラッシュROMやディスクなどの不揮発記憶デバイスである。また、不揮発メモリ 4 は、フラッシュメモリカードなどの取り外し可能なデバイス（メディア）、いわゆる IC カードであっても良い。CPU 1 は、これら RAM 2、デバイス 3 および不揮発メモリ 4 は、バス 5 を介して接続され、相互に通信可能な状態とされる。また、バス 5 は、ネットワークであっても良い。

【 0 0 1 2 】

この発明の機能別コンポーネントの初期化の制御を図 2 に示すフローチャートを参照して説明する。ステップ S 1 では、アプリケーションやデバイスドライバなどの機能別コンポーネント（以下、単にコンポーネントと称する）は、起動された際に、内部変数の初期化が行われる。ステップ S 2 では、コンポーネント自身によって状態保存データベースにコンポーネント間の依存関係が登録される。後述するシステムの状態保存処理の際には、この状態保存データベースに登録されているコンポーネントが状態保存の対象と見なされる。このように、この図 2 に示すフローチャートは、アプリケーションやデバイスドライバなどが起動した

時の初期化動作を示す。

【0013】

システムの状態保存処理を図3を参照して説明する。この図3は、CPU1からアクセス可能なメモリ上に存在し、CPU1が実行するモジュール間の関係を示している。アプリケーション14の中のあるアプリケーションから本モジュール11内のチェックポイントマネージャ12に対して、状態保存処理要求が出力される。チェックポイントマネージャ12は、アプリケーション14やデバイスドライバ15などのコンポーネントに対してスナップショット出力を行うように要求が出力される。この要求が出力される順序は、本モジュール11内の状態保存データベース13に登録されている順序に従う。この状態保存データベース13に登録されている順序がコンポーネント間の依存関係を表している。

【0014】

各コンポーネントがスナップショット出力の要求を受け取ると、チェックポイントマネージャ12を介して不揮発メモリドライバ16に対してコンポーネント毎の特定のアドレスに存在する関数をスナップショットデータとして出力する。各コンポーネントのスナップショットデータの出力が終了した時点で、出力された全てのスナップショットデータが図4に示すようなスナップショットファイルとされ、不揮発メモリドライバ16によって記録媒体（図示なし）に保存される。不揮発メモリドライバ16は、さまざまな記録媒体、例えば不揮発性メモリカードの書き込み／読み出しをサポートしている。そして、状態保存処理要求を出力したアプリケーション14に処理に戻る。

【0015】

スナップショットファイルは、図4に示すようにコンポーネント名（文字列）でタグ付けされ、例えばインデックスファイルと、データファイルとの組み合わせとして不揮発メモリドライバ16によって記録媒体に保存される。インデックスファイルのフォーマットの一例を図4Aに示し、データファイルのフォーマットの一例を図4Bに示す。図4Aに示すように、インデックスファイルには、コンポーネント名と、それに対応するデータのデータファイル中におけるオフセット位置が格納される。図4Bに示すように、データファイルには、各コンポーネ

ントが出力したスナップショットデータが1つのファイルに連結して格納される。

【0016】

このように、システムの状態保存処理は、アプリケーション実行中の任意のタイミングまたはシステムやアプリケーションから指定された周期毎に起動される。一度、状態保存処理が起動されると、その処理が終了するまで次の状態保存処理要求は待たされる。

【0017】

不揮発メモリドライバ16によって記録媒体に記録されるスナップショットファイルは、状態保存データベース13に登録されている順序に従って登録されているので、コンポーネント間の依存関係が反映されている。

【0018】

図5にシステムの状態保存処理の一例のフローチャートを参照して説明する。ステップS11では、スナップショットファイルの初期化が行われる。ステップS12では、状態保存データベース13がオープンされる。ステップS13では、状態保存の対象となるコンポーネントが存在するか否かが判断される。状態保存データベース13に登録されている順番に従って、まだ状態保存の対象となるコンポーネントが存在すると判断した場合、ステップS14へ制御が移り、状態保存の対象となるコンポーネントが存在しないと判断した場合、ステップS16へ制御が移る。

【0019】

ステップS14では、状態保存の対象となるコンポーネントからインデックスデータがチェックポイントマネージャ12を介して不揮発メモリドライバ16に出力される。ステップS15では、状態保存の対象となるコンポーネントからスナップショットデータがチェックポイントマネージャ12を介して不揮発メモリドライバ16に出力される。

【0020】

ステップS16では、不揮発メモリドライバ16によって記録媒体に格納されたスナップショットファイルがクローズされる。ステップS17では、保存状態

データベース 1 3 がクローズされる。

【 0 0 2 1 】

システムの状態復元処理を図 6 を参照して説明する。この図 6 は、CPU 1 からアクセス可能なメモリ上に存在し、CPU 1 が実行するモジュール間の関係を示している。システムのブート部 1 7 からチェックポイントマネージャ 1 2 に対して状態復元処理要求が出力される。この状態復元処理要求によって、チェックポイントマネージャ 1 2 は、不揮発メモリドライバ 1 6 に対してスナップショットファイルの読み込みを要求する。上述したようにスナップショットファイルは、コンポーネント間の依存関係が反映されている。チェックポイントマネージャ 1 2 は、スナップショットファイルに格納されたコンポーネント名から状態復元すべきアプリケーション・デバイスドライバ 1 8 のコンポーネントを検索し、まず必要であればそのコンポーネントを読み込み、起動する。

【 0 0 2 2 】

そして、コンポーネントが保持しているシンボル情報から呼び出すべき関数のアドレスを不揮発メモリドライバ 1 6 を介して記録媒体から取得し、そのアドレスから関数を呼び出す。呼び出される関数は、コンポーネント毎の特定のアドレスに存在するものである。これらの処理を、スナップショットファイルに保存されている全てのコンポーネントについて実行する。なおこの際、コンポーネント自身によって本モジュール 1 1 内の状態保存データベース 1 3 にコンポーネント間の依存関係が登録される。この状態保存データベース 1 3 に登録された内容は、以後の状態保存処理要求の際に参照される。

【 0 0 2 3 】

なお、状態復元処理は、システムのブート時だけでなく、任意のタイミングで可能である。ただし、一度、状態復元処理が起動されると、その状態復元処理が終了するまで、次の状態復元処理要求は待たされる。

【 0 0 2 4 】

図 7 にシステムの状態復元処理の一例のフローチャートを参照して説明する。ステップ S 2 1 では、不揮発メモリドライバ 1 6 によって記録媒体に格納されているスナップショットファイルがオープンされる。ステップ S 2 2 では、状態保

存データベース 13 がオープンされる。ステップ S 2 3 では、不揮発メモリドライバ 1 6 によって記録媒体に格納されているインデックスデータが読み込まれる。ステップ S 2 4 では、格納されているインデックスデータに基づいて、読み込む必要のあるコンポーネントは終了したか否かが判断される。インデックスデータに基づいて、まだ読み込む必要のあるコンポーネントの読み込みが終了していないと判断されると、ステップ S 2 5 へ制御が移り、読み込む必要のあるコンポーネントの読み込みが全て終了していると判断されると、ステップ S 2 9 へ制御が移る。

【 0 0 2 5 】

ステップ S 2 5 では、不揮発メモリドライバ 1 6 を介して記録媒体から格納されているスナップショットデータが読み込まれる。ステップ S 2 6 では、読み込まれたスナップショットデータに応じてコンポーネントが読み込まれ、起動される。ステップ S 2 7 では、状態復元処理が行われる。具体的には、コンポーネントが保持しているシンボル情報から読み出すべき関数のアドレスが取得され、そのアドレスから関数が呼び出される。呼び出される関数は、コンポーネント毎の特定のアドレスに存在するものである。ステップ S 2 8 では、コンポーネント自身によって本モジュール 1 1 内の状態保存データベース 1 3 にコンポーネント間の依存関係が登録される。そして、制御はステップ S 2 3 へ戻る。

【 0 0 2 6 】

ステップ S 2 9 では、不揮発メモリドライバ 1 6 によって記録媒体に格納されているスナップショットファイルがクローズされる。ステップ S 3 0 では、状態保存データベース 1 3 がクローズされる。

【 0 0 2 7 】

この実施形態では、システム上でアプリケーションが動作していても、そのシステムを操作している操作者が知ることなく、そのバックグラウンドで状態保存を実行することができる。

【 0 0 2 8 】

この実施形態は、図 1 に示す構成で有ればどのようなものにも適応することができる。例えば、携帯電話、PDA (Personal Digital Assistants)、ノート

PCなどに適用することができる。

【 0 0 2 9 】

この実施形態では、不揮発メモリドライバ16によってスナップショットファイルが格納されている記録媒体、例えばICカードを、スナップショットファイルを格納したPCと異なるPCに装着し、状態復元処理を行うと、スナップショットファイルを格納したPCと全く同じ環境で、ICカードを装着したPCを使用することができる。

【 0 0 3 0 】

【発明の効果】

この発明に依れば、電源の再投入時に、デバイスの依存関係に従いつつ起動し、高速に状態を復元することができる。

【 0 0 3 1 】

さらに、この発明に依れば、あるコンポーネントの状態復元のために必要なデータをコンポーネント自身が待避・復元するため、必要最小限の非常に少ないデータ量で実現することができる。

【 0 0 3 2 】

また、この発明に依れば、復元のための情報を携帯型メモリデバイスに保存することで、他の機器にそのメモリデバイスを接続すれば高速に状態を復元することができる。

【 0 0 3 3 】

さらに、この発明に依れば、システムのブート時に限らず、任意のタイミングで状態復元を行うことができる。

【図面の簡単な説明】

【図1】

この発明を適用することができるシステムのブロック図である。

【図2】

この発明のコンポーネントの初期化の一例を説明するためのフローチャートである。

【図3】

この発明のモジュール間の関係を説明するための略線図である。

【図 4】

この発明のスナップショットファイルのフォーマットの一例である。

【図 5】

この発明の状態保存処理の一例を説明するためのフローチャートである。

【図 6】

この発明のモジュール間の関係を説明するための略線図である。

【図 7】

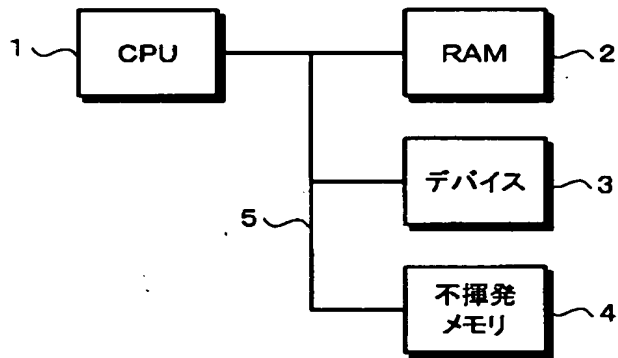
この発明の状態復元処理の一例を説明するためのフローチャートである。

【符号の説明】

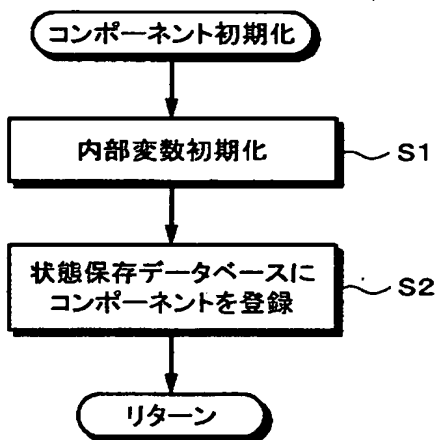
1 1・・・本モジュール、1 2・・・チェックポイントマネージャ、1 3・・・状態保存データベース、1 4・・・アプリケーション、1 5・・・ドライバ、1 6・・・不揮発メモリドライバ、1 7・・・ブート部、1 8・・・アプリケーション・ドライバ

【書類名】 図面

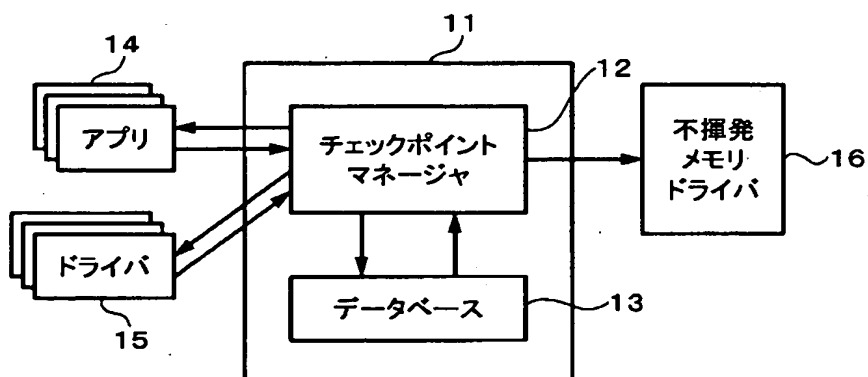
【図 1】



【図 2】



【図 3】



【図 4】

A

インデックスファイル

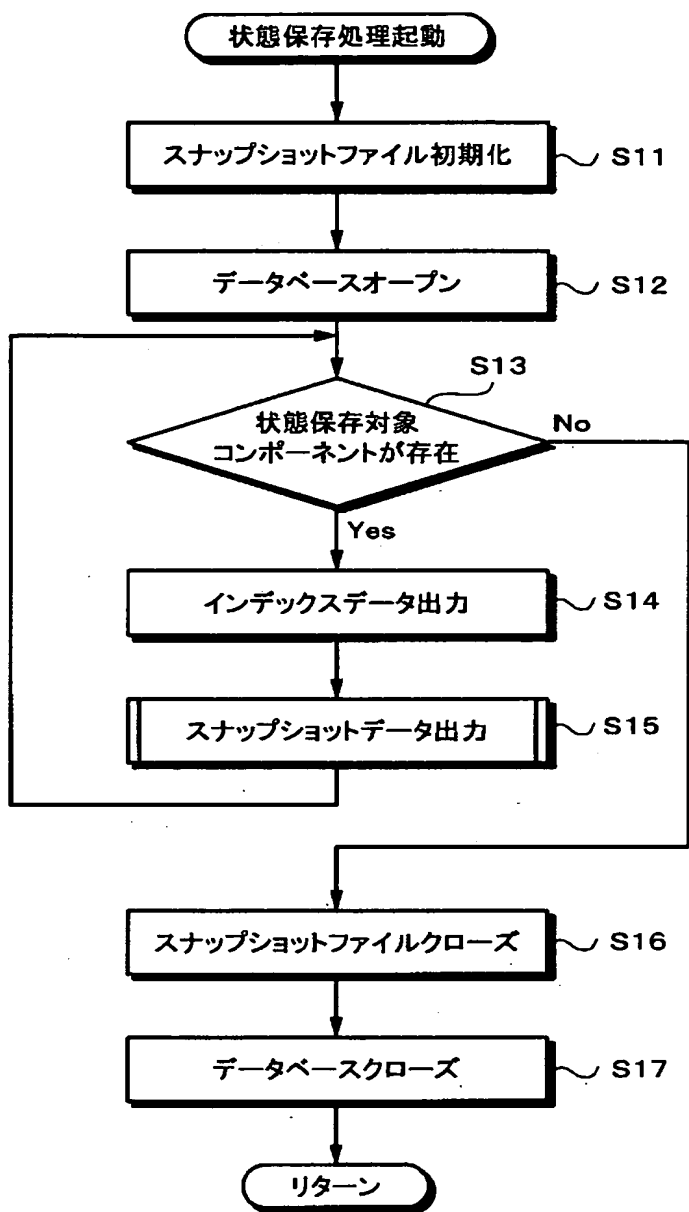
コンポーネント名 1	オフセット 1 <改行>
コンポーネント名 2	オフセット 2 <改行>
...	

B

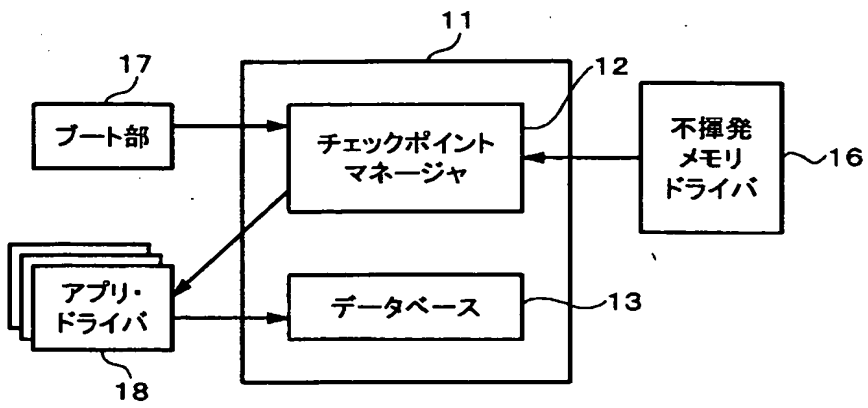
データファイル

<コンポーネント 1 状態データ>
<コンポーネント 2 状態データ>
...

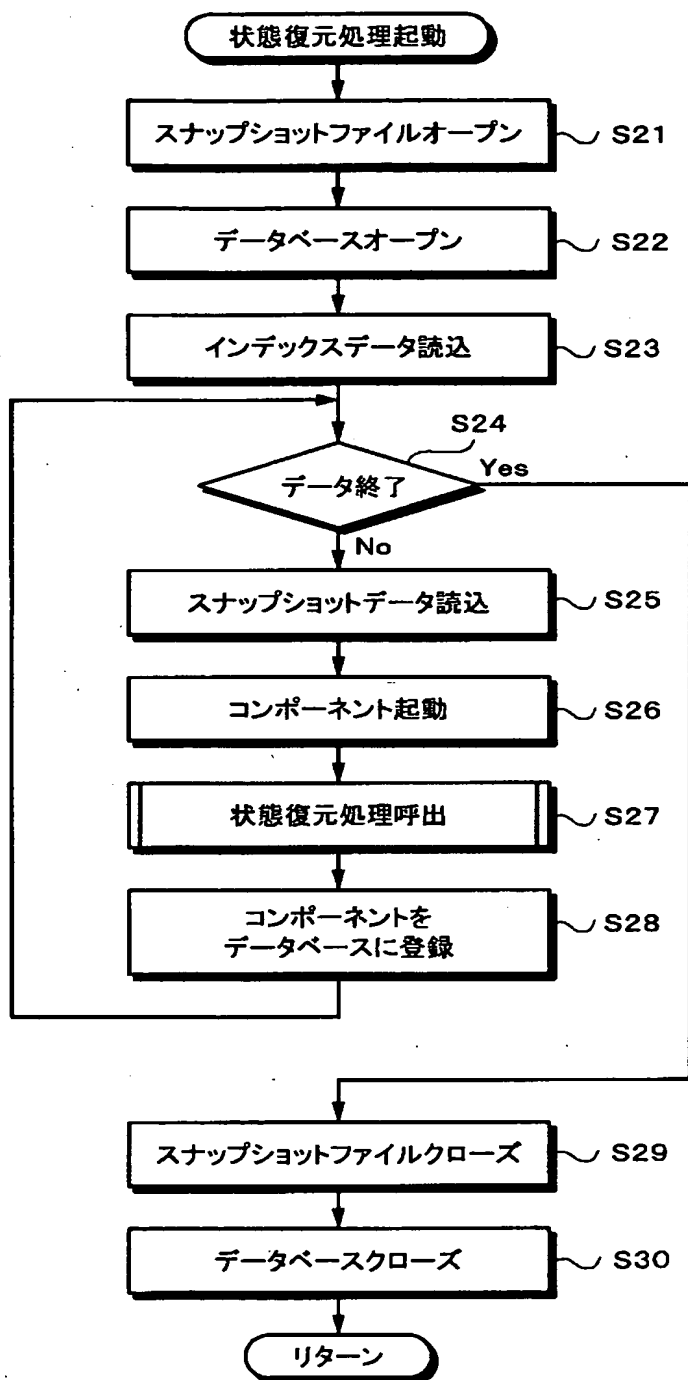
【図 5】



【図6】



【図 7】



【書類名】 要約書

【要約】

【課題】 システムに突発的な電源切断が発生しても瞬時に電源切断直前の状態に戻ることができる。

【解決手段】 アプリケーション 1 4 の中のあるアプリケーションから本モジュール 1 1 内のチェックポイントマネージャ 1 2 に対して、状態保存処理要求が出力される。マネージャ 1 2 は、アプリケーション 1 4 やデバイスドライバ 1 5 などのコンポーネントに対してスナップショット出力を行うように要求が出力される。このときの順序は、状態保存データベース 1 3 に登録されている順序に従い、その順序がコンポーネント間の依存関係を表している。各コンポーネントがスナップショット出力の要求を受け取ると、マネージャ 1 2 を介して不揮発メモリドライバ 1 6 に対してコンポーネント毎の特定のアドレスに存在する関数をスナップショットファイルとして出力する。

【選択図】 図 3

【書類名】 手続補正書
【提出日】 平成13年 1月23日
【あて先】 特許庁長官 及川 耕造 殿

【事件の表示】

【出願番号】 特願2000-357208

【補正をする者】

【識別番号】 000002185

【氏名又は名称】 ソニー株式会社

【代理人】

【識別番号】 100082762

【弁理士】

【氏名又は名称】 杉浦 正知

【電話番号】 03-3980-0339

【手続補正 1】

【補正対象書類名】 明細書

【補正対象項目名】 0 0 1 4

【補正方法】 変更

【補正の内容】 1

【手続補正 2】

【補正対象書類名】 要約書

【補正対象項目名】 全文

【補正方法】 変更

【補正の内容】 2

【プルーフの要否】 要

【 0 0 1 4 】

各コンポーネントがスナップショット出力の要求を受け取ると、チェックポイントマネージャ 1 2 を介して不揮発メモリドライバ 1 6 に対してコンポーネント毎の特定のアドレスに存在する関数が呼び出され、コンポーネントの状態をスナップショットデータとして出力する。各コンポーネントのスナップショットデータの出力が終了した時点で、出力された全てのスナップショットデータが図 4 に示すようなスナップショットファイルとされ、不揮発メモリドライバ 1 6 によって記録媒体（図示なし）に保存される。不揮発メモリドライバ 1 6 は、さまざまな記録媒体、例えば不揮発性メモリカードの書き込み／読み出しをサポートしている。そして、状態保存処理要求を出力したアプリケーション 1 4 に処理が戻る。

【書類名】 要約書

【要約】

【課題】 システムに突発的な電源切断が発生しても瞬時に電源切断直前の状態に戻ることができる。

【解決手段】 アプリケーション 1 4 の中のあるアプリケーションから本モジュール 1 1 内のチェックポイントマネージャ 1 2 に対して、状態保存処理要求が出力される。マネージャ 1 2 は、アプリケーション 1 4 やデバイスドライバ 1 5 などのコンポーネントに対してスナップショット出力を行うように要求が出力される。このときの順序は、状態保存データベース 1 3 に登録されている順序に従い、その順序がコンポーネント間の依存関係を表している。各コンポーネントがスナップショット出力の要求を受け取ると、マネージャ 1 2 を介して不揮発メモリドライバ 1 6 に対してコンポーネント毎の特定のアドレスに存在する関数が呼び出され、コンポーネントの状態をスナップショットファイルとして出力する。

【選択図】 図 3

認定・付加情報

特許出願の番号	特願 2000-357208
受付番号	50100088971
書類名	手続補正書
担当官	末武 実 1912
作成日	平成 13 年 1 月 26 日

<認定情報・付加情報>

【補正をする者】

【識別番号】

000002185

【住所又は居所】

東京都品川区北品川 6 丁目 7 番 35 号

【氏名又は名称】

ソニー株式会社

【代理人】

申請人

【識別番号】

100082762

【住所又は居所】

東京都豊島区南池袋二丁目 49 番 7 号 池袋パークビル 7 階

【氏名又は名称】

杉浦 正知

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日
[変更理由] 新規登録
住 所 東京都品川区北品川6丁目7番35号
氏 名 ソニー株式会社